Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Low-rank matrix recovery with total generalized variation for defending adversarial examples*

Wen LI[1,2], Hengyou WANG[†‡1,5], Lianzhi HUO[3], Qiang HE[1,5],
Linlin CHEN[1,5], Zhiquan HE[4], Wing W. Y. Ng[2]

[1]*School of Science, Beijing University of Civil Engineering and Architecture, Beijing 100044, China*
[2]*School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China*
[3]*Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China*
[4]*Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen 518060, China*
[5]*Institute of Big Data Modeling and Technology,*
Beijing University of Civil Engineering and Architecture, Beijing 100044, China
[†]E-mail: wanghengyou@bucea.edu.cn

Received Jan. 9, 2023; Revision accepted June 26, 2023; Crosschecked Feb. 19, 2024

Abstract: Low-rank matrix decomposition with first-order total variation (TV) regularization exhibits excellent performance in exploration of image structure. Taking advantage of its excellent performance in image denoising, we apply it to improve the robustness of deep neural networks. However, although TV regularization can improve the robustness of the model, it reduces the accuracy of normal samples due to its over-smoothing. In our work, we develop a new low-rank matrix recovery model, called LRTGV, which incorporates total generalized variation (TGV) regularization into the reweighted low-rank matrix recovery model. In the proposed model, TGV is used to better reconstruct texture information without over-smoothing. The reweighted nuclear norm and L_1-norm can enhance the global structure information. Thus, the proposed LRTGV can destroy the structure of adversarial noise while re-enhancing the global structure and local texture of the image. To solve the challenging optimal model issue, we propose an algorithm based on the alternating direction method of multipliers. Experimental results show that the proposed algorithm has a certain defense capability against black-box attacks, and outperforms state-of-the-art low-rank matrix recovery methods in image restoration.

Key words: Total generalized variation; Low-rank matrix; Alternating direction method of multipliers; Adversarial example
https://doi.org/10.1631/FITEE.2300017 **CLC number:** TP37

1 Introduction

Images are often polluted by noise, which affects image quality. With the evolution of deep neural network (DNN) research, it has been found that DNNs are also easily disturbed by adversarial noise. For example, if a subtle perturbation is added to the input image, the given DNN could be misclassified with high confidence. These visually imperceptible perturbations generate adversarial examples by attacking DNNs, which creates security risks in practical applications (Yuan et al., 2019; Dong XY et al., 2020; Mustafa et al., 2020; Wang YL et al., 2020; Xu J et al., 2020; Zhang et al., 2021).

In recent years, a variety of attack algorithms have been proposed for fooling DNNs. According

‡ Corresponding author
* Project supported by the National Natural Science Foundation of China (No. 62072024), the Outstanding Youth Program of Beijing University of Civil Engineering and Architecture, China (No. JDJQ20220805), and the Shenzhen Stability Support General Project (Type A), China (No. 20200826104014001)
ORCID: Wen LI, https://orcid.org/0009-0009-8206-9581; Hengyou WANG, https://orcid.org/0000-0001-6693-0161

to the principle of adversarial example generation, these attack methods can be grouped into two categories: gradient-based methods, such as the fast gradient sign method (FGSM) (Goodfellow et al., 2015) and project gradient descent (PGD) (Tabacof and Valle, 2016) methods, and optimization-based methods such as DeepFool (Moosavi-Dezfooli et al., 2016) and Carlini–Wagner (CW) (Carlini and Wagner, 2017) methods. Currently, more researchers are aiming to generate effective adversarial samples, but defense techniques are relatively backward.

In light of these attacks and to improve the robustness of the networks, a series of enhanced networks have been proposed. Reconstructing the input image, optimizing the target model, and adversarial training are the three main techniques of defense against attack. Because the fact that the defensive strategies of reconstructing input images do not alter the classification model, they have received widespread attention by researchers. They treat adversarial attacks as deliberate noise and eliminate the attack noise in the image. For example, Guo C et al. (2018) proposed four image transformations, i.e., bit-depth reduction, JPEG compression, total variation (TV) minimization, and image quilting. Bit-depth reduction removes adversarial perturbations on pixel values from an image via a simple type of quantization (Xu WL et al., 2017). TV minimization is a compressed sensing method that combines pixel dropout with TV regularization. Image quilting is a non-parametric technique that stitches small patches obtained from an image patch database to synthesize images (Efros and Freeman, 2001). PixelDefend (Song et al., 2018) moves the perturbed images back into the training data, and then cleans the images according to the distribution. Buckman et al. (2018) proposed a method to defend against adversarial examples using thermometer encoding.

It is well known that TV performs well at preserving local structural smoothness (Deng et al., 2013; Wang Q et al., 2018). It has been applied in defending adversarial examples. Recently, TV regularization has been incorporated into the low-rank matrix recovery model to solve the local smoothness problem of low-rank matrix image restoration. In terms of defense against adversarial examples, Zhao et al. (2021) proposed a new defense method combining the low-rank matrix theory and TV to eliminate adversarial attack noise. The main idea of low-rank matrix recovery is to decompose the observed image into the sum of a low-rank component and a sparse error component (Dong WS et al., 2013; Cao et al., 2015; Gu et al., 2017; Yang et al., 2018; Jing et al., 2019; Wu et al., 2019; Xie T et al., 2020; Zhan et al., 2020). The former is measured by the rank function to ensure the low-rank structure of the image. The latter is measured by an L_0-norm regularization term to guarantee its sparsity (Wen et al., 2015). However, it is difficult to obtain its optimal solution due to the nonlinearity of the rank function and the non-convexity of the L_0-norm. To solve this problem, Candès et al. (2011) proposed the robust principal component analysis (RPCA) algorithm, which relaxes the rank function to the nuclear norm and solves the nuclear norm minimization (NNM) problem using the singular value shrinkage algorithm. However, the above defense methods usually perform over-smoothing operations on the global image, causing significant damage to the non-attacked images. Therefore, these defense methods reduce the classification accuracy of the original (or non-attacked) images.

In traditional image denoising methods, to eliminate the negative effects of TV regularization, for example, over-smoothing or staircasing phenomena, Bredies et al. (2010) focused on the total generalized variation (TGV), which is higher-order TV regularization. They discovered that TGV is superior in image reconstruction. Based on this finding, Papafitsoros and Schönlieb (2014) proposed an image reconstruction algorithm by combining TV and second-order TGV. Guo WH et al. (2014) proposed a new detail-preserving regularization scheme, which combines TGV regularization with a wavelet transform model. The proposed model can better restore the edges and details of the image.

In this work, inspired by the above defense methods, a defense method based on low-rank matrix recovery with TGV regularization is proposed. The proposed model first makes full use of the low-rank matrix theory to explore the structure information, and further uses the local information retention ability of TGV to improve the image restoration performance. Thus, the proposed model not only defends against adversarial attacks but also guarantees the classification accuracy of the original images.

The main contributions can be summarized as follows:

1. A new method of defending adversarial examples based on low-rank matrix recovery with TGV is proposed. It not only removes the adversarial perturbation but also guarantees restoration of the edges and detailed information.

2. To deal with the challenging optimal model, an algorithm based on the alternating direction method of multipliers (ADMM) is designed. It divides the multi-variable optimization problem into several single-variable optimization sub-problems.

3. Experimental results demonstrate that our defense model greatly enhances the robustness of DNNs for adversarial examples, and exceeds state-of-the-art low-rank matrix restoration methods in image restoration.

2 Related works

In this work, we propose to use low-rank matrix recovery with TGV to defend against adversarial attacks in image classification tasks. Before specifying our approach, we review some related works. We focus our discussion on the low-rank matrix recovery algorithm and defense methods based on matrix recovery.

2.1 Low-rank matrix recovery algorithm

The general low-rank matrix recovery algorithm can be formulated as

$$\min_{X,E}(\|X\|_*+\lambda\|E\|_1)\ \text{s.t.}\ X^{\text{adv}}=X+E, \tag{1}$$

where $X^{\text{adv}}\in\mathbb{R}^{m\times n}$ is the input image, $\lambda>0$ is the regularization parameter, and $\|\cdot\|_*$ and $\|\cdot\|_1$ are the nuclear norm and L_1-norm of the matrix, respectively. The optimization problem can be solved by the principal component pursuit (PCP) algorithm (Candès et al., 2008). To further enhance the sparsity of the error matrix, the reweighted L_1-norm low-rank matrix recovery method (Deng et al., 2013) is proposed. This method aims to minimize the following objective function:

$$\min_{X,E}(\|X\|_*+\lambda\left\|W^E\odot E\right\|_1)\ \text{s.t.}\ X^{\text{adv}}=X+E, \tag{2}$$

where $W^E\in\mathbb{R}^{m\times n}$ is the weight of matrix E and "$\odot$" is the Hadamard product. Similar to the method of the reweighted L_1-norm minimization problem, the low-rank matrix recovery based on reweighted nuclear norm is formulated as

$$\begin{aligned}&\min_{X,E}\Big(\sum_{j=1}^{n}w_j^X\sigma_j+\lambda\left\|W^E\odot E\right\|_1\Big)\\&\text{s.t. } X^{\text{adv}}=X+E,\end{aligned} \tag{3}$$

where σ_j is the j^{th} singular value of matrix X, $W^X=[w_j^X]$ is the weight of σ_j, and $\sum_{j=1}^{n}w_j^X\sigma_j$ is a weighted matrix nuclear norm. It can be optimized by the non-uniform singular value thresholding (NSVT) method (Peng et al., 2014).

To significantly improve the quality of the recovered image, the smoothed and reweighted low-rank matrix recovery (SRLRMR) method (Wang HY et al., 2018) is proposed. The method incorporates the TV norm into the reweighted low-rank matrix analysis to achieve structural smoothness:

$$\begin{aligned}&\min_{X,E}\Big(\sum_{j=1}^{n}w_j^X\sigma_j+\lambda\left\|W^E\odot E\right\|_1+\eta\|X\|_{\text{TV}}\Big)\\&\text{s.t. } \Delta_{\text{L}}\le x_{i,j}\le\Delta_{\text{U}},\quad X^{\text{adv}}-X=E.\end{aligned} \tag{4}$$

The constraints Δ_{L} and Δ_{U} represent the lower and upper bounds for each pixel, respectively. For example, in an image, pixel values are in the range $[0,255]$.

2.2 Defense methods based on matrix recovery

TV minimization (Guo C et al., 2018) randomly selects pixels using Bernoulli sampling. It retains the pixel value when $M(i,j)=1$, and then uses TV minimization to construct an image X. The reconstructed image is similar to the input image (adversarial example) X^{adv} on the selected set of pixels, which can be described as the following problem:

$$\min_X\Big(\left\|(1-M)\odot(X-X^{\text{adv}})\right\|_{\text{F}}^2+\lambda_{\text{TV}}\text{TV}_p(X)\Big), \tag{5}$$

where $\|\cdot\|_{\text{F}}$ is the Frobenius norm, and $\text{TV}_p(X)$ represents the TV of L_p-norm in X, defined as

$$\begin{aligned}\text{TV}_p(X)=&\sum_{i=2}^{N}\|X(i,:)-X(i-1,:)\|_p\\&+\sum_{j=2}^{N}\|X(:,j)-X(:,j-1)\|_p.\end{aligned} \tag{6}$$

In the experiments, set $p=2$; objective function (5) is a convex function about X, which makes the solution easy. TV measures the amount of variation

in the neighborhood of pixels in image X, enhancing the smoothness of local regions. Its results show that TV minimization can help remove adversarial perturbations in images.

In another defense work, combined with the low-rank matrix theory, Zhao et al. (2021) proposed a new defense method via low-rank completion of high-sensitivity points (LCHP) to remove adversarial noise. The method uses the gradient information back-propagation to the input image by the network to determine the high-sensitivity points, which is an important contribution to image classification. Then it divides the image pixels into two groups by setting appropriate thresholds: high-sensitivity pixels and low-sensitivity pixels. Existing image smoothing methods based on TV minimization are used for the reconstruction of low-sensitivity pixels. For the high-sensitivity key points, the low-rank completion method is proposed to preserve structure information.

3 The proposed method

3.1 Overview

A DNN can be easily destroyed by adversarial noise. To eliminate the adversarial noise, we first destroy many adversarial noise structures by randomly masking pixels in the image, and the pixel values of the masked part are set to zero. Then, the low-rank matrix recovery method is adopted to reconstruct the image and the intrinsic structure information from the unmasked pixels. As we know, natural images always have the characteristics of a global structure. For example, a cat image takes "cat" as the main structure. Thus, the inherent structure of the image can be restored by the low-rank matrix recovery theory. In addition, the edges and local detail information can be better restored by TGV regularization. Therefore, the reconstructed images have wonderful quality for classification.

The entire defense process is shown in Fig. 1. Specifically, the pixels of images in the dataset are randomly masked with probability p to obtain the masked images. Then, the LRTGV regularization algorithm is applied to obtain the reconstructed image. Finally, the reconstructed images can be classified correctly with higher probability by the trained DNN.

3.2 Low-rank matrix recovery with TGV

Due to the difficulty in choosing an appropriate threshold, TV (i.e., first-order TV regularization) suffers from the issue of over-smoothing, which dramatically decreases the classification accuracy of clean images. In comparison, TGV (i.e., high-order TV regularization) can better preserve image details. Inspired by TGV, we present the LRTGV to defend the adversarial samples. The proposed method can effectively remove adversarial noise while preserving the global structure and local detail information of the image, to realize defense of adversarial examples. This method solves the over-smoothing problem of first-order TV regularization and improves the classification accuracy of the original images. The LRTGV regularization model can be formulated as follows:

$$\begin{aligned}&\min_{X,E}\left(\sum_{j=1}^{n} w_j^X\sigma_j+\lambda\left\|W^E\odot\hat{M}\odot E\right\|_1+\mathrm{TGV}_\alpha^2(X)\right)\\&\text{s.t. } X^{\mathrm{adv}}=X+E,\end{aligned}\tag{7}$$

where $\hat{M}$ is the binary mask matrix of the noise location and λ is the balance parameter for the three items in the objective function. If there is a noisy pixel at position (i,j), $\hat{M}(i,j)$ is set as 0; $\hat{M}(i,j)=1$ otherwise.

TGV (Bredies et al., 2010) is a generalization of the TV regularization with higher order ($\geq 2^{\mathrm{nd}}$ order), which is similar to a nonlinear method. We

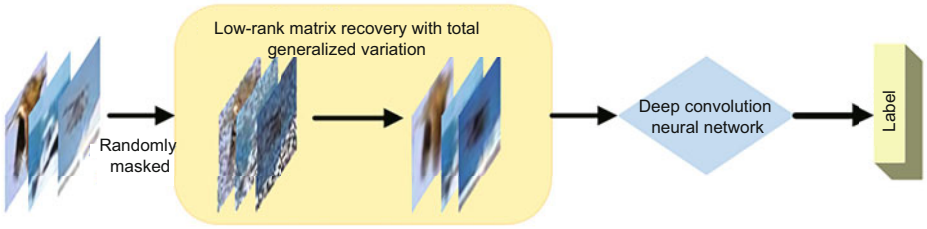

Fig. 1 Training process of low-rank matrix recovery with total generalized variation (LRTGV)

introduce the second-order TGV (noted as $\mathrm{TGV}^2_{\tilde{\alpha}}$) in our work; for more detailed information about TGV, refer to Bredies et al. (2010). To simplify the calculation formula, the $\mathrm{TGV}^2_{\tilde{\alpha}}$ regularization term can be described as

$$\mathrm{TGV}^2_{\tilde{\alpha}}(u)=\min_{p\in\Omega}\Big(\tilde{\alpha}_1\int_{\Omega}|\nabla u-p|\mathrm{d}x+\tilde{\alpha}_0\int_{\Omega}|G(p)|\mathrm{d}x\Big), \tag{8}$$

where $G(p)=\frac{1}{2}(\nabla p+\nabla p^{\mathrm{T}})$ is the symmetric derivative. $\mathrm{TGV}^2_{\tilde{\alpha}}$ can be further described as

$$\mathrm{TGV}^2_{\tilde{\alpha}}(u)=\min_{p}(\tilde{\alpha}_1\|\nabla u-p\|_1+\tilde{\alpha}_0\|G(p)\|_1). \tag{9}$$

To simplify the formula and calculation, we approximate Cu to ∇u, and let C_1 and C_2 be the first-order forward finite difference matrices, representing periodic boundary conditions in the horizontal and vertical directions, respectively. Therefore, the definitions of C and $G(p)$ are as follows:

$$C=\begin{bmatrix} C_1 \\ C_2 \end{bmatrix},$$
$$G(p)=\begin{bmatrix} C_1p_1 & \frac{1}{2}(C_2p_1+C_1p_2) \\ \frac{1}{2}(C_2p_1+C_1p_2) & C_2p_2 \end{bmatrix}, \tag{10}$$

where $p=(p_1,p_2)$. Then Eq. (9) can be described as

$$\mathrm{TGV}^2_{\tilde{\alpha}}(u)=\min_{p}(\tilde{\alpha}_1\|Cu-p\|_1+\tilde{\alpha}_0\|G(p)\|_1). \tag{11}$$

In Eq. (11), the first term on the right side corresponds to the total size of the first-order discrete gradients in the vertical and horizontal directions, and the second term corresponds to the total size of the second-order discrete gradients. $\tilde{\alpha}_0\in(0,1)$ and $\tilde{\alpha}_1\in(0,1)$ are the weight parameters.

3.3 Optimization algorithm

Note that incorporating TGV regularization into the low-rank matrix recovery model makes the problem more challenging to solve. The mathematical solutions developed in existing methods (Wang Q et al., 2018) can no longer be used to solve this new optimization problem. In this subsection, we adopt the ADMM to solve the optimization problem (7). Finally, the LRTGV algorithm is presented. To solve the problem easily, the auxiliary variable A is introduced and the objective function can be reformulated as follows:

$$\begin{aligned}&\min_{X,A,E}\Big(\sum_{j=1}^{n}w_j^X\sigma_j+\lambda\left\|W^E\odot\hat{M}\odot E\right\|_1+\mathrm{TGV}^2_{\alpha}(A)\Big)\\ &\text{s.t. } X^{\mathrm{adv}}=X+E,\quad X=A.\end{aligned} \tag{12}$$

Thus, the augmented Lagrangian function of function (12) is constructed as follows:

$$\begin{aligned}&f(X,A,E,Y_1,Y_2)\\ &=\sum_{j=1}^{m}w_j^X\sigma_j+\|W^E\odot\hat{M}\odot E\|_1+\mathrm{TGV}^2_{\alpha}(A)\\ &\quad+\left\langle Y_1,X^{\mathrm{adv}}-X-E\right\rangle+\langle Y_2,A-X\rangle\\ &\quad+\frac{\mu}{2}\left\|X^{\mathrm{adv}}-X-E\right\|_{\mathrm{F}}^2+\frac{\mu}{2}\|A-X\|_{\mathrm{F}}^2,\end{aligned} \tag{13}$$

where $Y_1,Y_2\in\mathbb{R}^{m\times n}$ are Lagrangian multipliers, μ is a penalty parameter, and $<,>$ is the inner product operation of two matrices. For the above Lagrangian function, it is difficult to analyze its closed-form solution. We will provide an iterative solution based on the ADMM. We divide problem (13) into several sub-problems. In our problem, there are five main variables X,A,E,Y_1,Y_2, which should be updated in each iteration. In the following, we explain how each of these five variables will be updated.

1. Optimizing variable X

For fixed variables A,E,Y_1,Y_2, the updating formula of variable X can be obtained by solving the X sub-problem as follows:

$$\begin{aligned}X&=\arg\min_X f(X,A,E,Y_1,Y_2)\\ &=\arg\min_X\Big(\sum_{j=1}^{m}w_j^X\sigma_j+\left\langle Y_1,X^{\mathrm{adv}}-X-E\right\rangle\\ &\quad+\langle Y_2,A-X\rangle+\tfrac{\mu}{2}\left\|X^{\mathrm{adv}}-X-E\right\|_{\mathrm{F}}^2\\ &\quad+\tfrac{\mu}{2}\|A-X\|_{\mathrm{F}}^2\Big)\\ &=\arg\min_X\Big(\sum_{j=1}^{m}w_j^X\sigma_j+Y_1^{\mathrm{T}}(X^{\mathrm{adv}}-X-E)\\ &\quad+\tfrac{\mu}{2}\left(X^{\mathrm{adv}}-E-X\right)^{\mathrm{T}}\left(X^{\mathrm{adv}}-E-X\right)\\ &\quad+Y_2^{\mathrm{T}}(A-X)+\tfrac{\mu}{2}(A-X)^{\mathrm{T}}(A-X)\Big)\\ &\overset{\mathrm{a}}{=}\arg\min_X\Big(\sum_{j=1}^{m}w_j^X\sigma_j-Y_1^{\mathrm{T}}X-Y_2^{\mathrm{T}}X+\mu X^{\mathrm{T}}X\\ &\quad-\mu(X^{\mathrm{adv}}-E)^{\mathrm{T}}X-\mu A^{\mathrm{T}}X\Big)\\ &=\arg\min_X\Big(\sum_{j=1}^{m}w_j^X\sigma_j+\mu X^{\mathrm{T}}X\\ &\quad-\mu(X^{\mathrm{adv}}+A-E+(Y_1+Y_2)/\mu)^{\mathrm{T}}X\Big)\\ &=\arg\min_X\Big(\sum_{j=1}^{m}w_j^X\sigma_j\\ &\quad+\mu\left\|X-\tfrac{1}{2}(X^{\mathrm{adv}}+A-E+(Y_1+Y_2)/\mu)\right\|_{\mathrm{F}}^2\Big).\end{aligned} \tag{14}$$

The equality "$\overset{\mathrm{a}}{=}$" of Eq. (14) can be derived by removing the constant term of the optimization problem on variable X. If we consider X as a variable, the other fixed variables A,E,Y_1,Y_2

can be regarded as constants, which do not affect the solution of the optimization problem. Let $L=\frac{1}{2}(X^{\mathrm{adv}}+A-E+(Y_1+Y_2)/\mu)$, Eq. (14) can be solved through NSVT, and the optimal solution can be expressed as $X=\mathcal{D}_{\mu^{-1}W^X}(L)$; $\mathcal{D}(\cdot)$ is an NSVT operator (Peng et al., 2014).

2. Optimizing variable A

For fixed variables X,E,Y_1,Y_2, the iterative formula on variable A can be denoted as

$$\begin{aligned}A&=\arg\min_A f(X,A,E,Y_1,Y_2)\\&=\arg\min_A\big(\mathrm{TGV}^2_{\tilde{\alpha}}(A)+\langle Y_2,A-X\rangle\\&\quad+\tfrac{\mu}{2}\|A-X\|_{\mathrm{F}}^2\big)\\&=\arg\min_X\big(\mathrm{TGV}^2_{\tilde{\alpha}}(X)+Y_2^{\mathrm{T}}(A-X)\\&\quad+\tfrac{\mu}{2}(A-X)^{\mathrm{T}}(A-X)\big)\\&\overset{\mathrm{b}}{=}\arg\min_A\big(\mathrm{TGV}^2_{\tilde{\alpha}}(A)+Y_2^{\mathrm{T}}A+\tfrac{\mu}{2}A^{\mathrm{T}}A-\mu X^{\mathrm{T}}A\big)\\&=\arg\min_A\big(\mathrm{TGV}^2_{\tilde{\alpha}}(A)\\&\quad+\tfrac{\mu}{2}(A^{\mathrm{T}}A-2(X-Y_2/\mu)^{\mathrm{T}}A)\big)\\&=\arg\min_A\big(\mathrm{TGV}^2_{\tilde{\alpha}}(A)+\tfrac{\mu}{2}\|A-(X-Y_2/\mu)\|_{\mathrm{F}}^2\big)\\&=\arg\min_A\big(\mathrm{TGV}^2_{\tilde{\alpha}}(A)+\tfrac{\mu}{2}\|A-Z\|_{\mathrm{F}}^2\big),\end{aligned}\tag{15}$$

where $Z=A-Y_2/\mu$. Similar to the solution of variable X, when the other variables X,E,Y_1,Y_2 are fixed, the equality "$\overset{\mathrm{b}}{=}$" of Eq. (15) can be derived by removing the irrelevant term of the optimization problem on variable A.

To solve the TGV-based regularization models, inspired by Bredies et al. (2010), we transform problem (15) to the standard form of the TGV denoising problem:

$$\begin{aligned}A&=\arg\min_A\big(\mathrm{TGV}^2_{\tilde{\alpha}}(A)+\tfrac{\mu}{2}\|A-Z\|_{\mathrm{F}}^2\big)\\&=\arg\min_A\big(\tfrac{2}{\mu}\mathrm{TGV}^2_{\tilde{\alpha}}(A)+\|A-Z\|_{\mathrm{F}}^2\big)\\&=\arg\min_A\big(\tfrac{2}{\mu}(\tilde{\alpha}_1\|CA-p\|_1+\tilde{\alpha}_0\|G(p)\|_1)\\&\quad+\|A-Z\|_{\mathrm{F}}^2\big)\\&=\arg\min_A\big((\tfrac{2\tilde{\alpha}_1}{\mu}\|CA-p\|_1+\tfrac{2\tilde{\alpha}_0}{\mu}\|G(p)\|_1)\\&\quad+\|A-Z\|_{\mathrm{F}}^2\big)\\&=\arg\min_A\big((\alpha_1\|CX-p\|_1+\alpha_0\|G(p)\|_1)\\&\quad+\|A-Z\|_{\mathrm{F}}^2\big),\end{aligned}\tag{16}$$

where $\alpha_0=\frac{2\tilde{\alpha}_0}{\mu}$ and $\alpha_1=\frac{2\tilde{\alpha}_1}{\mu}$. So, problem (16) can be solved as the $\mathrm{TGV}^2_{\alpha}-L^2$ image denoising model (Bredies et al., 2010).

3. Optimizing variable E

For fixed variables X,A,Y_1,Y_2, the objective function of variable E can be described as

$$\begin{aligned}E&=\arg\min_E f(X,A,E,Y_1,Y_2)\\&=\arg\min_E\Big(\lambda\Big\|W^E\odot\hat{M}\odot E\Big\|_1\\&\quad+\langle Y_1,X^{\mathrm{adv}}-X-E\rangle+\tfrac{\mu}{2}\big\|X^{\mathrm{adv}}-X-E\big\|_{\mathrm{F}}^2\Big)\\&=\arg\min_E\Big(\lambda\Big\|W^E\odot\hat{M}\odot E\Big\|_1+Y_1^{\mathrm{T}}(X^{\mathrm{adv}}-X-E)\\&\quad+\tfrac{\mu}{2}\left(X^{\mathrm{adv}}-X-E\right)^{\mathrm{T}}\left(X^{\mathrm{adv}}-X-E\right)\Big)\\&\overset{\mathrm{c}}{=}\arg\min_E\Big(\lambda\Big\|W^E\odot\hat{M}\odot E\Big\|_1-{Y_1}^{\mathrm{T}}E+\tfrac{\mu}{2}E^{\mathrm{T}}E\\&\quad-\mu(X^{\mathrm{adv}}-X)^{\mathrm{T}}E\Big)\\&=\arg\min_E\Big(\lambda\Big\|W^E\odot\hat{M}\odot E\Big\|_1\\&\quad+\tfrac{\mu}{2}(E^{\mathrm{T}}E-2(X^{\mathrm{adv}}-X+Y_1/\mu)^{\mathrm{T}}E)\Big)\\&=\arg\min_E\Big(\lambda\Big\|W^E\odot\hat{M}\odot E\Big\|_1\\&\quad+\tfrac{\mu}{2}\left\|E-\left(X^{\mathrm{adv}}-X+Y_1/\mu\right)\right\|_{\mathrm{F}}^2\Big)\\&=\arg\min_E\Big(\lambda\Big\|W^E\odot\hat{M}\odot E\Big\|_1+\tfrac{\mu}{2}\|E-N\|_{\mathrm{F}}^2\Big).\end{aligned}\tag{17}$$

The equality "$\overset{\mathrm{c}}{=}$" of problem (17) can be obtained as the above problems by removing the terms that are irrelevant to E. Let $N=X^{\mathrm{adv}}-X+Y_1/\mu$. The optimal solution is $E=\mathcal{S}_{\lambda\mu^{-1}W^E}(N)$; $\mathcal{S}(\cdot)$ is the non-uniform soft thresholding iterative shrinking operator (Candès et al., 2008).

4. Optimizing variables Y_1 and Y_2

Y_1 and Y_2 are Lagrangian multipliers, and they can be updated as follows:

$$Y_1=Y_1+\mu\left(X^{\mathrm{adv}}-X-E\right),\tag{18}$$

$$Y_2=Y_2+\mu\left(A-X\right).\tag{19}$$

The algorithm for solving the inner optimization problem of LRTGV is summarized in Algorithm 1, which obtains the low-rank matrix X by ADMM. Although the solutions have been obtained for each variable, the masked matrix $\hat{M}(i,j)$ could be unknown or known. If $\hat{M}$ is unknown, we consider $\hat{M}$ as a matrix whose entries are all ones. In this case, we can directly apply Algorithm 1 to restore images, which is a matrix recovery problem. In contrast, the task becomes simpler when $\hat{M}$ is known. We use multiple iterations to complete the matrix, which is a classic matrix completion problem summarized in Algorithm 2. In Section 4.2, we perform image restoration for different $\hat{M}$ cases, including the cases in which $\hat{M}$ is known or unknown. For defense adversarial examples in Section 4.3, because we randomly masked the adversarial noise, the locations of the

masked pixel points are known. So, $\hat{M}$ is known and we defend adversarial examples using Algorithm 2.

3.4 Algorithm complexity analysis

We analyze the complexity of our proposed algorithm. When $X^{\text{adv}} \in \mathbb{R}^{m\times n}$ is the input image, the computational complexity of the proposed method consists mainly of three parts: (1) solving the X sub-problem, (2) solving the A sub-problem, and (3) solving the E sub-problem.

The major computation of solving the X sub-problem is the singular value decomposition (SVD) of matrix X^{adv}. The complexity of SVD for matrix X^{adv} is $O(mn^2)$. The complexity of solving the A sub-problem is $O(mn)$. The E sub-problem is solved by a soft thresholding iterative shrinking operator, whose complexity is $O(mn)$. The complexity of one iteration in Algorithm 1 is $O(\max(mn^2, mn)) = O(mn^2)$. Therefore, the complexity of Algorithm 1 is $O(kmn^2)$, and the complexity of our proposed algorithm is the same as those of many current algorithms. When M is known, we use Algorithm 2 to further improve the restoration performance. The complexity of Algorithm 2 is $O(tkmn^2)$, where k and t are the numbers of iterations of the inner and outer layers, respectively. The maximum of t is 3, which indicates that the complexity of our overall algorithm does not increase significantly.

Certainly, our proposed method includes an extra addition operation when compared to the NSVT method. Thus, we test the runtime of our method on the Lenna image in Fig. 2. We resize the image from 100×100 to 900×900 in 100 steps. We compare the time complexity of our algorithm against existing low-rank recovery methods, including the PCP algorithm, the reweighted L_1 algorithm, the NSVT method, and the SRLRMR method. As shown in Fig. 3, the runtime of our proposed method is only slightly higher than that of other methods with the dimension of the matrix increased.

Algorithm 1 ADMM of LRTGV

Input: X^{adv}, λ, μ, α_0, α_1
Init: $A_0 \in \mathbb{R}^{m\times n}$, $E_0 \in \mathbb{R}^{m\times n}$, $X_0 \in \mathbb{R}^{m\times n}$, $Y_{1,0}, Y_{2,0} \in \mathbb{R}^{m\times n}$, $\mu = \mu_0$, $\delta = 1.1$, and maxiter = 100
while $\left\|X^{\text{adv}} - X_{k+1} - E_{k+1}\right\|_{\text{F}} / \left\|X^{\text{adv}}\right\|_{\text{F}} > \tau$ and $k <$ maxiter **do**
1. $L = \frac{1}{2}(X^{\text{adv}} + A - E + (Y_1 + Y_2)/\mu)$. Then update X_{k+1} based on Eq. (14) via the NSVT operator $X = \mathcal{D}_{\mu^{-1}W^X}(L)$
2. Let $Z_{k+1} = X_{k+1} - Y_{2,k}/\mu_k$. Using α_0 and α_1, then update A based on Eq. (16) via the $\text{TGV}_\alpha^2 - L^2$ denoising model
3. Let $N_{k+1} = X^{\text{adv}} - X_{k+1} + Y_{1,k}/\mu_k$. Then $E_{k+1} = \mathcal{S}_{\lambda\mu^{-1}W^E}(N_{k+1})$ based on Eq. (17)
4. Update $Y_{1,k+1}, Y_{2,k+1}$ via Eqs. (18) and (19), respectively
5. $\mu_{k+1} = \delta\mu_k$
6. $k = k + 1$
end
Output: optimal X^*

Algorithm 2 LRTGV with known mask $\hat{M}$

Input: X^{adv}, $\hat{M}$, λ, μ, α_0, α_1
Init: $X_0 \in \mathbb{R}^{m\times n}$, $E_0 \in \mathbb{R}^{m\times n}$, outmaxiter = 3, $X^{a'} = X^{\text{adv}} \odot \hat{M}$. Compute the low-rank matrix $X_1 = U\Sigma V$ and sparse error matrix E_1 of observation matrix X^{adv}. Let $w_j^X = \frac{1}{(\text{diag}(\Sigma))_j + \varepsilon_X}$, $w_j^E = \frac{1}{|E_{1,j}| + \varepsilon_E}$
while $t <$ outmaxiter **do**
1. Use Algorithm 1 to calculate reconstruction matrix X_t^* of $X^{a'}$
2. $X^{a'} = X^{\text{adv}} \odot \hat{M} + X_t^* \odot (1 - \hat{M})$
3. $t = t + 1$
end
Output: optimal $X = X_t^*$

4 Experimental results and analysis

In this section, to verify the performance of the proposed LRTGV algorithm for image recovery and defense ability, we conduct several experiments on real-world databases to analyze the properties of the proposed algorithm. We not only use the proposed method to remove the adversarial noise, but also restore the image corrupted with salt and pepper noise. To show its superiority, we compare it with the latest existing methods.

4.1 Experimental setup

To coordinate with the experimental environment of the comparison methods, the image restoration experiments were performed in a MATLAB environment using a computer with Intel core i5, 2.5 GHz CPU, 8 GB RAM, and Microsoft Windows 10. For the experiments involving defending adversarial examples, we used the PyTorch framework to implement experiments and evaluated the defensive effects under black-box attacks. The algorithms were performed on an Ubuntu 18.04 system with an NVIDIA RTX 2080Ti GPU and 64 GB RAM.

In our algorithm, there were four parameters λ, μ, α_0, and α_1. Similar to Wang HY et al. (2018), λ was set to $1\Big/\sqrt{\max(m,n)}$ and μ was set to $c_1/\left\|X^{\text{adv}}\right\|_{\text{F}}$, where m and n are the dimensions

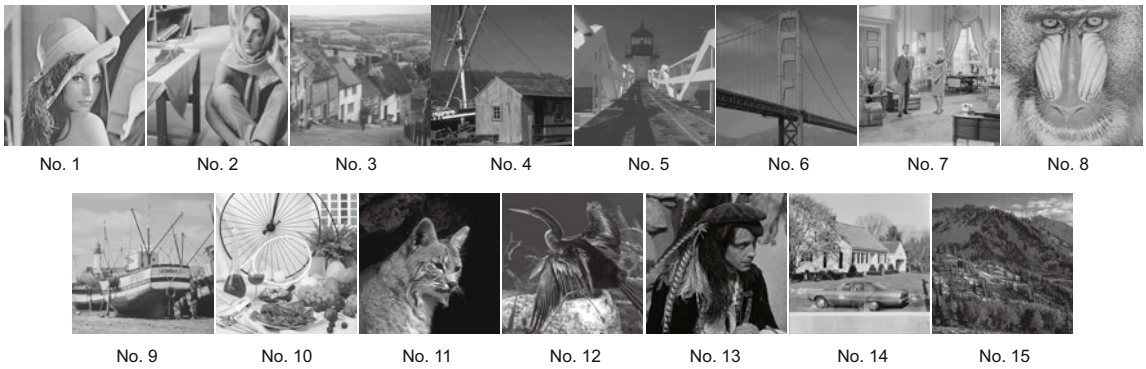

Fig. 2 The 15 images used in the experiments

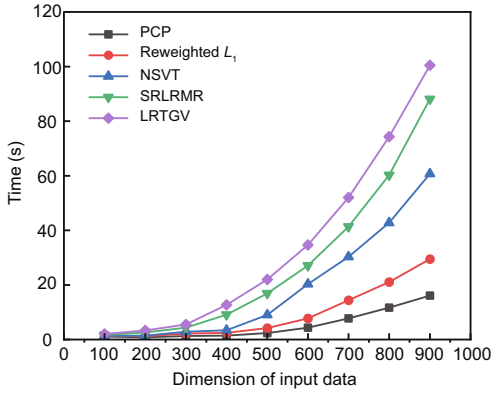

Fig. 3 Runtime comparison

of the input image. Thus, the effect of parameter μ is equivalent to that of parameter c_1. In the following experiments, we verified the influence of other parameters on the recovery effect. We tested the peak signal-to-noise ratio (PSNR) of different images with the change of c_1 for several classical grayscale images randomly selected in Fig. 2. As shown in Fig. 4a, the best experimental results were obtained with $c_1 = 3.75$.

Inspired by the TGV-denoising algorithm (Bredies et al., 2010), we assumed $\alpha_0 = 2v \in (0,1)$ and $\alpha_1 = v \in (0,1)$. Therefore, studying the influence of α_0 and α_1 could be converted to studying parameter v. The value of v is inversely correlated with the size of the data matrix X^{adv} and the parameter μ, so we set $v = \frac{2c_2}{\mu\sqrt{m}} \times 10^{-4}$. The suitable value of c_2 was selected by the experiments, and we found that the best results were obtained at $c_2 = 2.5$ in Fig. 4b. In summary, all parameters were set as follows: $\lambda = 1\Big/\sqrt{\max(m,n)}$, $\mu = 3.75/\left\|X^{\mathrm{adv}}\right\|_{\mathrm{F}}$, $\alpha_0 = \frac{1}{\mu\sqrt{m}} \times 10^{-3}$, and $\alpha_1 = \frac{5}{\mu\sqrt{m}} \times 10^{-4}$.

4.2 Image restoration

We evaluated the LRTGV algorithm's capability to restore images from normal noise. The 15 grayscale images in Fig. 2 were selected for image

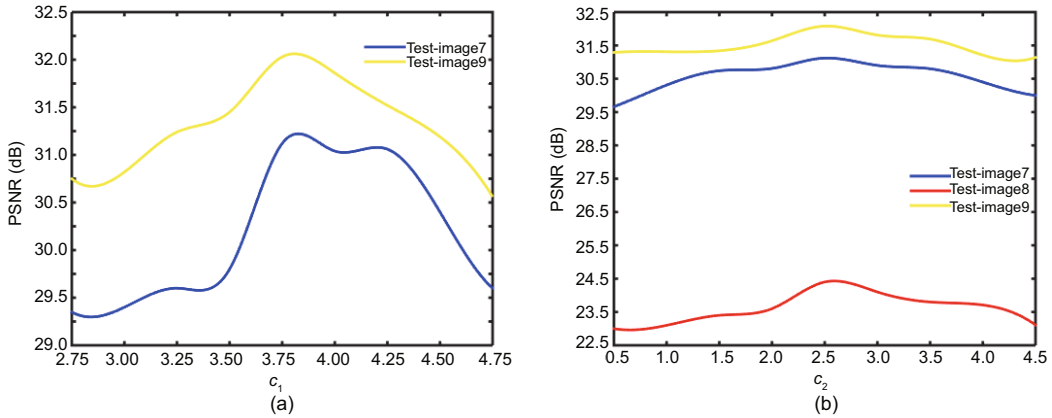

Fig. 4 Peak signal-to-noise ratio (PSNR) with c_1 (a) and c_2 (b)

restoration, and salt and pepper noise with different densities was added. Fig. 5 shows that 20% salt and pepper noise was added to the original image. To evaluate the performance of our proposed algorithm, we compared it with the PCP algorithm (Candès et al., 2008), reweighted L_1 algorithm (Deng et al., 2013), NSVT algorithm (Peng et al., 2014), ROUTE algorithm (Guo XJ and Lin, 2018), and SRLRMR algorithm (Wang HY et al., 2018).

Table 1 shows the PSNR values of the 15 reconstructed images. When 20% salt and pepper noise was added to the original image, the average PSNR value of the SRLRMR algorithm is improved by 1.01 dB compared with that of our proposed algorithm. Introducing TGV into the SRLRMR model can preserve textures and sharp edges. For the image with relatively complex details (image number 2 in Fig. 2), compared with our algorithm, the PSNR value of SRLRMR was improved by 2.51 dB; for the image with relatively simple details (image number 1 in Fig. 2), the PSNR value was improved by 1.95 dB. Fig. 5 shows the restoration visualization results of the given images with 20% salt and pepper noise. Restoration visualization results of LRTGV are compared with those of the five mentioned algorithms.

These experimental results demonstrate that the proposed method has better robustness to the image restoration with salt and pepper noise, and can better preserve the edge and local detail information of the image. We need to emphasize that if our LRTGV model does not contain the TGV regularization, it degenerates into the NSVT method. Therefore, our proposed method has a competitive advantage in image restoration.

It should be noted that the results of LRTGV in Table 1 are restored only by Algorithm 1. In this case, the noise location matrix is considered unknown, so the mask matrix is set as $\hat{M}(i,j)=1$. In addition, to verify Algorithm 2, we fixed the noise location and completed the matrix with LRTGV. Now, the noise location matrix is considered known. Specifically, if there is a noisy pixel at position (i,j), $\hat{M}(i,j)$ is set as 0; $\hat{M}(i,j)=1$ otherwise. As shown in Fig. 6, the recovery performance on each image is further improved.

4.3 Defense adversarial examples

We compare our proposed LRTGV adversarial defense algorithm with state-of-the-art defense

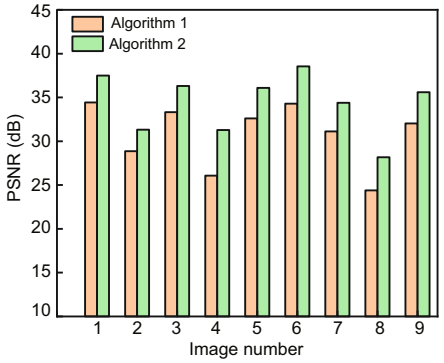

Fig. 6 Peak signal-to-noise ratio (PSNR) comparison of restoration between Algorithms 1 and 2 (References to color refer to the online version of this figure)

Fig. 5 Reconstruction results of the six algorithms for images with 20% salt and pepper noise: (a) original; (b) noisy; (c) PCP; (d) reweighted L_1; (e) NSVT; (f) ROUTE; (g) SRLRMR; (h) LRTGV

Table 1 PSNR comparison of PCP, reweighted L_1, NSVT, ROUTE, SRLRMR, and our LRTGV algorithms

Noise density	Image No.	PSNR (dB)						ΔPSNR (dB)
		PCP	Reweighted L_1	NSVT	ROUTE	SRLRMR	LRTGV	LRTGV−SRLRMR
$p=0.2$	1	26.75	23.84	32.15	28.50	32.47	**34.42**	+1.95
	2	24.23	22.33	26.28	23.44	26.36	**28.87**	+2.51
	3	28.41	26.75	31.99	29.22	**33.71**	33.32	−0.39
	4	25.90	22.79	28.10	25.52	**29.64**	26.08	−3.56
	5	29.77	26.43	34.63	29.56	**34.88**	32.61	−2.27
	6	29.98	27.96	34.01	28.79	33.73	**34.30**	+0.57
	7	25.71	23.49	28.78	26.18	29.31	**31.12**	+1.81
	8	21.72	20.43	22.12	20.00	23.71	**24.40**	+0.69
	9	25.22	23.01	27.65	25.31	28.35	**32.03**	+3.68
	10	18.89	18.93	**19.33**	16.94	19.13	19.23	+0.10
	11	26.34	26.29	28.28	27.60	29.88	**31.26**	+1.38
	12	24.13	24.09	25.20	23.70	25.55	**28.27**	+2.72
	13	24.09	24.02	26.20	25.04	27.86	**29.04**	+1.18
	14	24.05	24.00	26.50	25.77	27.99	**32.71**	+4.72
	15	22.67	22.53	23.10	19.95	23.58	**23.63**	+0.05
	Average	25.19	23.79	27.62	25.03	28.41	**29.42**	+1.01
$p=0.3$	1	21.27	22.17	23.84	23.52	30.67	**33.45**	+2.78
	2	19.10	20.85	20.77	20.43	24.35	**25.72**	+1.37
	3	23.00	25.42	25.56	26.07	31.15	**32.20**	+1.05
	4	22.52	23.04	26.61	23.26	27.36	**27.53**	+0.17
	5	25.20	26.69	28.73	27.41	33.31	**33.65**	+0.34
	6	27.29	26.57	30.57	28.31	33.01	**33.99**	+0.98
	7	20.92	22.01	22.93	22.85	27.60	**29.39**	+1.79
	8	18.09	19.21	18.12	18.00	21.90	**22.96**	+1.06
	9	21.25	21.62	22.01	21.94	27.66	**29.98**	+2.32
	10	16.43	16.40	17.56	14.82	**18.12**	17.61	−0.51
	11	21.41	21.43	22.00	23.92	25.31	**31.83**	+6.52
	12	19.61	19.14	21.34	20.82	22.25	**26.79**	+4.54
	13	19.22	18.82	22.14	21.94	23.22	**28.41**	+5.19
	14	20.11	19.91	20.54	21.76	26.83	**29.14**	+2.31
	15	18.23	18.11	20.43	17.73	21.34	**22.49**	+1.15
	Average	20.91	21.43	22.88	22.19	26.27	**28.34**	+2.07

Best results are in bold. PSNR: peak signal-to-noise ratio

methods based on image noise removal, including JPEG compression (Guo C et al., 2018), bit-depth reduction (Guo C et al., 2018), TV minimization (Guo C et al., 2018), image quilting (Guo C et al., 2018), PixelDefend (Song et al., 2018), and LCHP (Zhao et al., 2021). The above methods are similar to our proposed method. All methods aim to remove adversarial attack noise and restore the original image. We evaluate these defense methods against black-box attacks, including PGD, momentum, and diverse-input attack methods. The FGSM is the first gradient-based attack method. Dong YP et al. (2018) proposed an iterative attack method to boost adversarial attacks with momentum. To further improve the success rate of black-box attacks, Xie CH et al. (2019) introduced input diversity based on MI-FGSM, and proposed the diverse input iterative FGSM. The method improved the transferability of adversarial examples.

The experiments were performed on the CIFAR-10 and SVHN datasets for performance evaluation and comparison. The CIFAR-10 dataset consists of 60 000 images in 10 classes with an image size of 32×32. SVHN is a street number image dataset, which contains 73 257 training images and 26 032 test images. In our experiments, we used the following method to generate masks $\hat{M}$ with different reserving probabilities: For each image, we selected 10 masks in total with reserving probability p ranging from 0.6 to 0.8 with an equal interval of 0.02, i.e., 0.62, 0.64, $\cdots$. For test images, we randomly sampled one mask with reserving probability $p=0.7$, which can be used

as known pixels. Thus, in the following experiments, mask matrix $\hat{M}$ is known, so we complete the image by Algorithm 2.

The CIFAR-10 dataset was trained on the ResNet-18 model for image classification and achieved a classification accuracy of 87.06%. Table 2 summarizes the defense results against black-box attacks. Compared with the above attack methods, our proposed LRTGV method showed a stronger defense ability. Except for JPEG compression and bit-depth reduction, our LRTGV method allowed the least damage to the original clean image. However, the performance of JPEG declines sharply with larger adversarial noise. Our LRTGV method can more effectively defend against larger attack noise. The LRTGV algorithm shows strong generalization ability in black-box attack experiments. Fig. 7 shows the visualization effect on recovery images using different defense methods in the CIFAR-10 dataset, and the recovery effect on adversarial examples is shown in Fig. 8. Because the algorithm is related to TGV, comparison methods that are closely related to TV regularization were chosen, such as TV minimization and LCHP. From the visual effect of the restored image, it can be found that under the same loss probability, TV minimization cannot reconstruct the image very well, while the image restored by LCHP is too smooth. The LRTGV method retains certain image detail and has a good defensive performance.

We also conducted experiments on the SVHN dataset. The baseline accuracy of the SVHN dataset on the ResNet-18 model is 96.82%. Table 3 summarizes the classification accuracy of various defense methods. From the numerical results, it can be concluded that our method outperforms other methods when the perturbation is large, and can effectively remove the adversarial noise.

In addition, we replaced the LRTGV module in Fig. 1 with the above low-rank matrix recovery methods, such as PCP, reweighted L_1, and NSVT methods, for defending against adversarial examples. We conducted experiments on the CIFAR-10 dataset to analyze the algorithm components. As seen from Table 4, compared with the NSVT method, our

Table 2 Black-box defense on CIFAR-10

Method	Classification accuracy (%)			
	Clean	PGD ($\varepsilon = 8/255$ / $16/255$)	Momentum ($\varepsilon = 8/255$ / $16/255$)	Diverse-input ($\varepsilon = 8/255$ / $16/255$)
No defense	87.06	34.18 / 21.30	24.36 / 12.16	20.81 / 10.39
JPEG	**85.22**	48.03 / 30.05	34.75 / 15.78	30.14 / 13.49
Bit-depth reduction	85.02	44.50 / 28.38	32.54 / 15.43	28.28 / 13.52
TV minimization	82.71	50.47 / 32.57	36.63 / 16.15	32.36 / 14.07
Image quilting	77.49	64.76 / 52.86	53.61 / 34.12	51.25 / 30.51
PixelDefend	78.88	46.9 / 29.12	34.86 / 16.00	31.09 / 13.78
LCHP	77.60	62.20 / 56.32	53.86 / 45.36	51.02 / 44.46
LRTGV	83.85	**71.45 / 62.06**	**67.17 / 52.19**	**61.39 / 50.84**

Best results are in bold. ε represents the maximum perturbation magnitude of the attack methods. No defence indicates the reference classification trained on ResNet-18

Table 3 Black-box defense on SVHN

Method	Classification accuracy (%)			
	Clean	PGD ($\varepsilon = 8/255$ / $16/255$)	Momentum ($\varepsilon = 8/255$ / $16/255$)	Diverse-input ($\varepsilon = 8/255$ / $16/255$)
No defense	96.82	30.18 /13.62	29.96 / 9.99	28.79 / 9.29
JPEG	93.75	30.76 / 13.98	30.35 / 10.24	29.43 / 9.58
Bit-depth reduction	92.66	33.19 / 15.51	32.63 /11.36	31.58 / 10.68
TV minimization	92.59	34.27 / 15.32	33.61 / 11.06	32.71 /10.36
Image quilting	93.30	36.04 / 16.68	35.14 / 11.6	34.71 / 11.12
PixelDefend	92.54	30.90 /13.56	30.63 / 9.89	29.38 / 9.28
LCHP	91.56	57.87 / 29.81	**60.56** / 25.17	**60.39** / 24.23
LRTGV	**95.99**	**58.89 / 31.35**	60.17 / **25.90**	59.82 / **25.12**

Best results are in bold. ε represents the maximum perturbation magnitude of the attack methods. No defence indicates the reference classification trained on ResNet-18

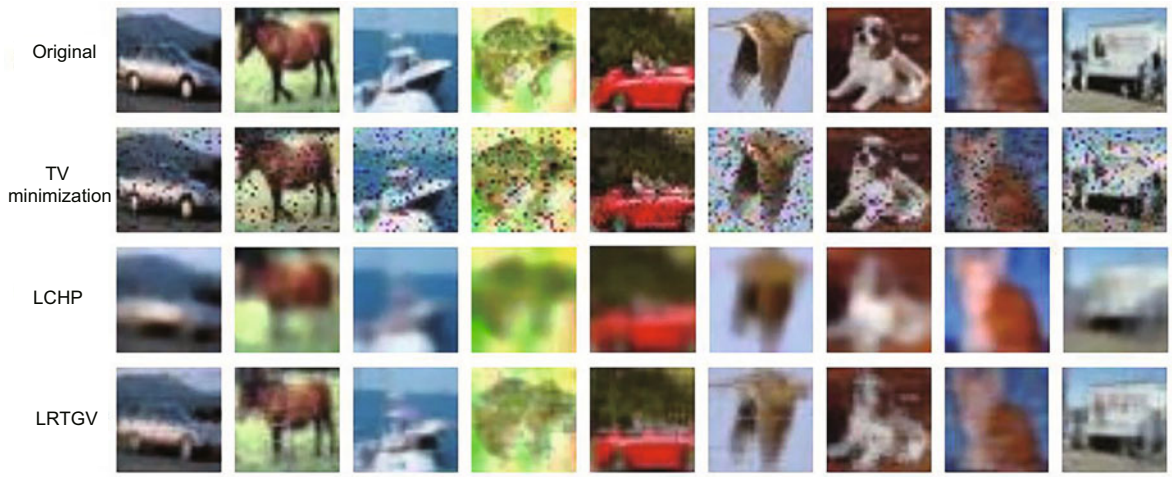

Fig. 7 CIFAR-10 image with low-rank matrix recovery with total generalized variation (LRTGV)

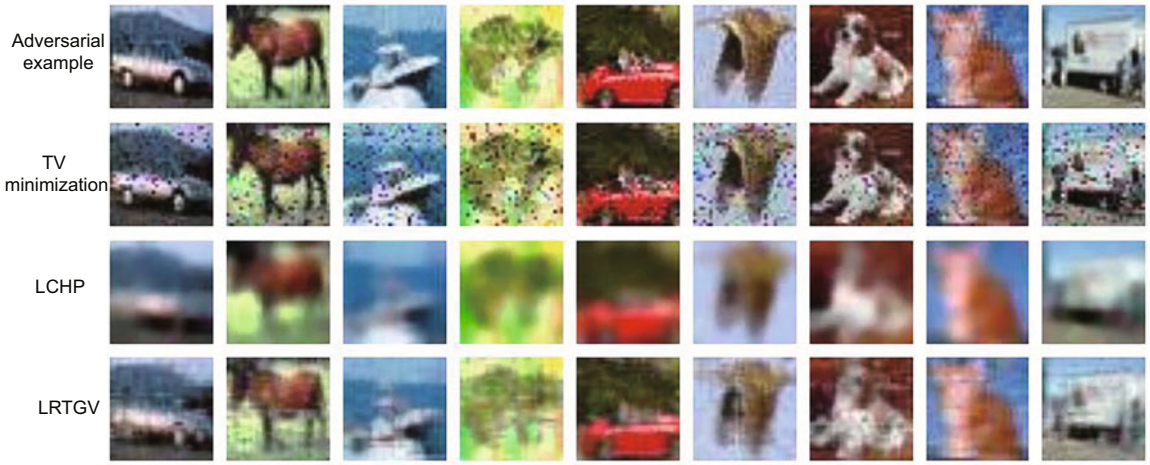

Fig. 8 CIFAR-10 adversarial example with low-rank matrix recovery with total generalized variation (LRTGV)

algorithm is advantageous in terms of accuracy on both clean and adversarial examples. As a result, our LRTGV method surpasses state-of-the-art low-rank matrix recovery methods in image restoration and improves the accuracy of neural network classification of adversarial examples. It is essential to introduce TGV to the low-rank restoration model to form LRTGV.

Table 4 Analysis of algorithm components on the CIFAR-10 dataset

Method	Classification accuracy (%)		
	Clean	PGD	Momentum
PCP	79.08	53.52	39.17
Reweighted L_1	79.32	55.36	44.65
NSVT	80.55	60.64	51.47
LRTGV	83.85	71.45	67.17

5 Conclusions

In this paper, to overcome the disadvantage of the first-order TV regularization denoising method, we integrated TGV regularization into the reweighted low-rank matrix decomposition model to remove the adversarial noise. It can be used to defend against different types of adversarial attacks. To address the proposed optimization problem, an iterative solution based on the alternating direction method of multipliers was designed, which can be applied to effectively eliminate adversarial noise. Experimental results showed that our proposed model consistently outperforms state-of-the-art baselines in image restoration and defense attacks, and improves the overall robustness under various adversarial attacks.

Contributors

All the authors designed the research. Wen LI and Hengyou WANG proposed the main idea. Wen LI performed the experiments and drafted the paper. Lianzhi HUO, Qiang HE, and Linlin CHEN helped organize the paper. Hengyou WANG, Zhiquan HE, and Wing W. Y. Ng revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Bredies K, Kunisch K, Pock T, 2010. Total generalized variation. *SIAM J Imag Sci*, 3(3):492-526. https://doi.org/10.1137/090769521

Buckman J, Roy A, Raffel C, et al., 2018. Thermometer encoding: one hot way to resist adversarial examples. 6th Int Conf on Learning Representations.

Candès EJ, Wakin MB, Boyd SP, 2008. Enhancing sparsity by reweighted l_1 minimization. *J Fourier Anal Appl*, 14(5-6):877-905. https://doi.org/10.1007/s00041-008-9045-x

Candès EJ, Li XD, Ma Y, et al., 2011. Robust principal component analysis? *J ACM*, 58(3):11. https://doi.org/10.1145/1970392.1970395

Cao FL, Cai MM, Tan YP, 2015. Image interpolation via low-rank matrix completion and recovery. *IEEE Trans Circ Syst Video Technol*, 25(8):1261-1270. https://doi.org/10.1109/TCSVT.2014.2372351

Carlini N, Wagner D, 2017. Towards evaluating the robustness of neural networks. IEEE Symp on Security and Privacy, p.39-57. https://doi.org/10.1109/SP.2017.49

Deng Y, Dai QH, Liu RS, et al., 2013. Low-rank structure learning via nonconvex heuristic recovery. *IEEE Trans Neur Netw Learn Syst*, 24(3):383-396. https://doi.org/10.1109/TNNLS.2012.2235082

Dong WS, Zhang L, Shi GM, et al., 2013. Nonlocally centralized sparse representation for image restoration. *IEEE Trans Image Process*, 22(4):1620-1630. https://doi.org/10.1109/TIP.2012.2235847

Dong XY, Han JF, Chen DD, et al., 2020. Robust superpixel-guided attentional adversarial attack. IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.12892-12901. https://doi.org/10.1109/CVPR42600.2020.01291

Dong YP, Liao FZ, Pang TY, et al., 2018. Boosting adversarial attacks with momentum. IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.9185-9193. https://doi.org/10.1109/CVPR.2018.00957

Efros AA, Freeman WT, 2001. Image quilting for texture synthesis and transfer. Proc 28th Annual Conf on Computer Graphics and Interactive Techniques, p.341-346. https://doi.org/10.1145/383259.383296

Goodfellow IJ, Shlens J, Szegedy C, 2015. Explaining and harnessing adversarial examples. https://arxiv.org/abs/1412.6572

Gu SH, Xie Q, Meng DY, et al., 2017. Weighted nuclear norm minimization and its applications to low level vision. *Int J Comput Vis*, 121(2):183-208. https://doi.org/10.1007/s11263-016-0930-5

Guo C, Rana M, Cisse M, et al., 2018. Countering adversarial images using input transformations. https://arxiv.org/abs/1711.00117

Guo WH, Qin J, Yin WT, 2014. A new detail-preserving regularization scheme. *SIAM J Imag Sci*, 7(2):1309-1334. https://doi.org/10.1137/120904263

Guo XJ, Lin ZC, 2018. Low-rank matrix recovery via robust outlier estimation. *IEEE Trans Image Process*, 27(11):5316-5327. https://doi.org/10.1109/TIP.2018.2855421

Jing PG, Su YT, Nie LQ, et al., 2019. A framework of joint low-rank and sparse regression for image memorability prediction. *IEEE Trans Circ Syst Video Technol*, 29(5):1296-1309. https://doi.org/10.1109/TCSVT.2018.2832095

Moosavi-Dezfooli SM, Fawzi A, Frossard P, 2016. DeepFool: a simple and accurate method to fool deep neural networks. IEEE Conf on Computer Vision and Pattern Recognition, p.2574-2582. https://doi.org/10.1109/CVPR.2016.282

Mustafa A, Khan SH, Hayat M, et al., 2020. Image super-resolution as a defense against adversarial attacks. *IEEE Trans Image Process*, 29:1711-1724. https://doi.org/10.1109/TIP.2019.2940533

Papafitsoros K, Schönlieb CB, 2014. A combined first and second order variational approach for image reconstruction. *J Math Imag Vis*, 48(2):308-338. https://doi.org/10.1007/s10851-013-0445-4

Peng YG, Suo JL, Dai QH, et al., 2014. Reweighted low-rank matrix recovery and its application in image restoration. *IEEE Trans Cybern*, 44(12):2418-2430. https://doi.org/10.1109/TCYB.2014.2307854

Song Y, Kim T, Nowozin S, et al., 2018. PixelDefend: leveraging generative models to understand and defend against adversarial examples. https://arxiv.org/abs/1710.10766

Tabacof P, Valle E, 2016. Exploring the space of adversarial images. Int Joint Conf on Neural Networks, p.426-433. https://doi.org/10.1109/IJCNN.2016.7727230

Wang HY, Cen YG, He ZQ, et al., 2018. Reweighted low-rank matrix analysis with structural smoothness for image denoising. *IEEE Trans Image Process*, 27(4):1777-1792. https://doi.org/10.1109/TIP.2017.2781425

Wang Q, Wu ZJ, Jin J, et al., 2018. Low rank constraint and spatial spectral total variation for hyperspectral image mixed denoising. *Signal Process*, 142:11-26. https://doi.org/10.1016/j.sigpro.2017.06.012

Wang YL, Wu KL, Zhang CS, 2020. Adversarial attacks on deep unfolded networks for sparse coding. IEEE Int Conf on Acoustics, Speech and Signal Processing, p.5974-5978. https://doi.org/10.1109/ICASSP40776.2020.9054671

Wen JM, Li DF, Zhu FM, 2015. Stable recovery of sparse signals via l_p-minimization. *Appl Comput Harmon Anal*, 38(1):161-176. https://doi.org/10.1016/j.acha.2014.06.003

Wu HC, Xiao L, Lian ZC, et al., 2019. Locally low-rank regularized video stabilization with motion diversity constraints. *IEEE Trans Circ Syst Video Technol*, 29(10):2873-2887. https://doi.org/10.1109/TCSVT.2018.2875671

Xie CH, Zhang ZS, Zhou YY, et al., 2019. Improving transferability of adversarial examples with input diversity. IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.2725-2734. https://doi.org/10.1109/CVPR.2019.00284

Xie T, Li ST, Sun B, 2020. Hyperspectral images denoising via nonconvex regularized low-rank and sparse matrix decomposition. *IEEE Trans Image Process*, 29:44-56. https://doi.org/10.1109/TIP.2019.2926736

Xu J, Li YM, Jiang Y, et al., 2020. Adversarial defense via local flatness regularization. IEEE Int Conf on Image Processing, p.2196-2200. https://doi.org/10.1109/ICIP40778.2020.9191346

Xu WL, Evans D, Qi YJ, 2017. Feature squeezing: detecting adversarial examples in deep neural networks. https://arxiv.org/abs/1704.01155

Yang S, Luo B, Li CL, et al., 2018. Fast grayscale-thermal foreground detection with collaborative low-rank decomposition. *IEEE Trans Circ Syst Video Technol*, 28(10):2574-2585. https://doi.org/10.1109/TCSVT.2017.2721460

Yuan XY, He P, Zhu QL, et al., 2019. Adversarial examples: attacks and defenses for deep learning. *IEEE Trans Neur Netw Learn Syst*, 30(9):2805-2824. https://doi.org/10.1109/TNNLS.2018.2886017

Zhan SH, Wu JG, Han N, et al., 2020. Group low-rank representation-based discriminant linear regression. *IEEE Trans Circ Syst Video Technol*, 30(3):760-770. https://doi.org/10.1109/TCSVT.2019.2897072

Zhang YC, Li HR, Zheng Y, et al., 2021. Enhanced DNNs for malware classification with GAN-based adversarial training. *J Comput Virol Hack Tech*, 17(2):153-163. https://doi.org/10.1007/S11416-021-00378-Y

Zhao ZQ, Wang HY, Sun H, et al., 2021. Removing adversarial noise via low-rank completion of high-sensitivity points. *IEEE Trans Image Process*, 30:6485-6497. https://doi.org/10.1109/TIP.2021.3086596